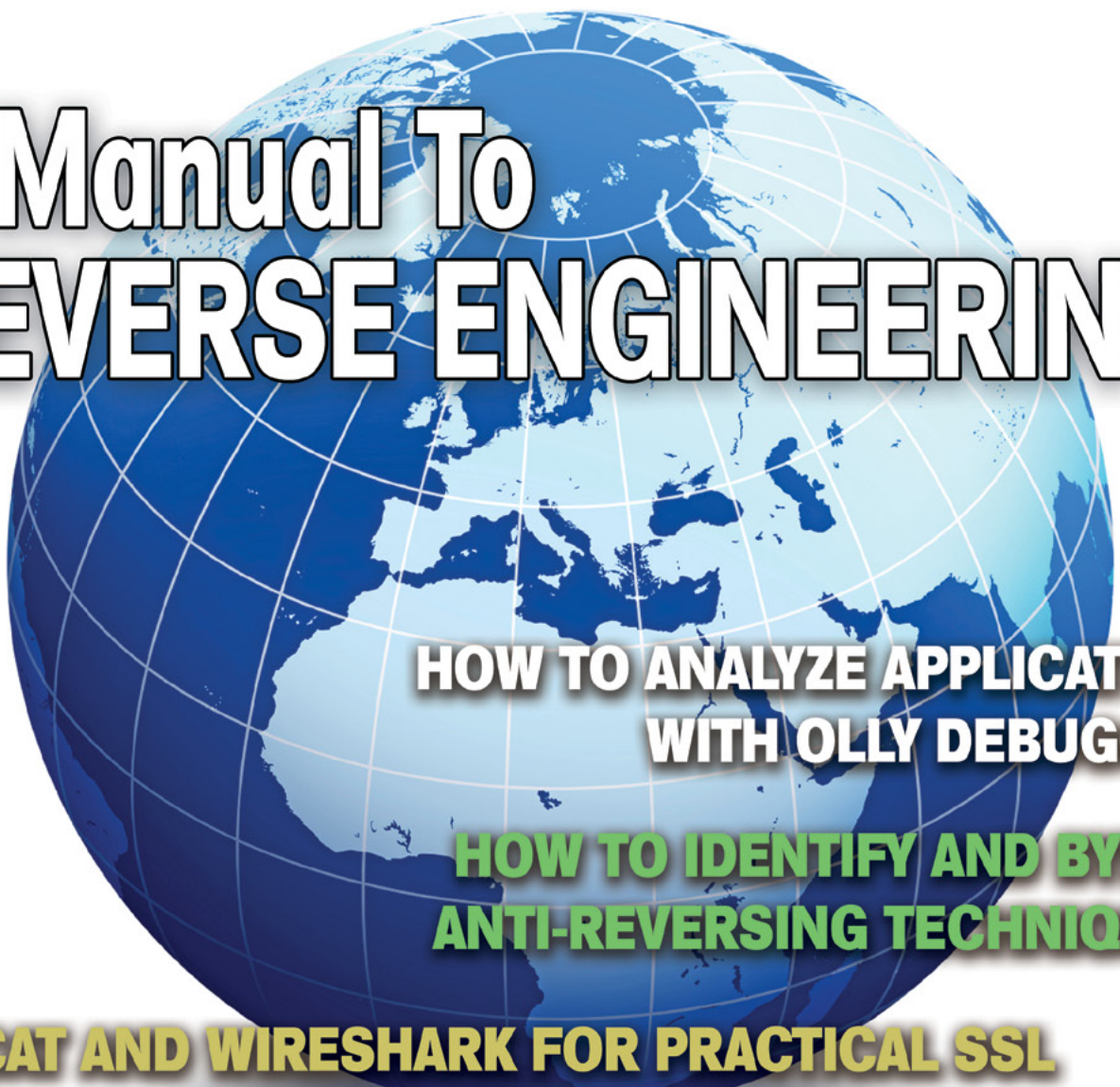
Hakin9

Exploiting Software DIGEST

Vol.2 No.10
Issue 10/2012(14) ISSN: 1733-7186



A Manual To REVERSE ENGINEERING

HOW TO ANALYZE APPLICATIONS WITH OLLY DEBUGGER?

HOW TO IDENTIFY AND BYPASS ANTI-REVERSING TECHNIQUES?

SOCAT AND WIRESHARK FOR PRACTICAL SSL PROTOCOL REVERSE ENGINEERING

DISASSEMBLE AND DEBUG EXECUTABLE PROGRAMS ON LINUX, WINDOWS AND MAC OS X

PLUS

JSCRAMBLER – PROTECT YOUR CODE (REVIEW)
MODERN WEBSITES, WHICH USE WEB 2.0 AND AJAX, OFTEN GENERATE HTML AND JAVASCRIPT CODE ON THE FLY

|w| http://www.sysmoth.com
|e| info@sysmoth.com
|p| +92 333 2319192

sysmoth

Cloud & Virtualization | Server Administration | Security & Compliance

Cloud & Virtualization

- Cloud & Virtualization Consultancy
- Building Virtualized Infrastructure
- Infrastructure on Public Cloud
- Building Private Cloud
- Cloud Management Setups
- Big Data Setups
- Infrastructure Management and Support

Server Administration

- Server Setups
- Control Panels Setups
- Server/Network Monitoring Setups
- Site Migration
- Server Optimization
- Email Setups
- Version Control Setups
- Server Automation
- Server Management & Support
- Load Balancing, FailOver and
- Geo Distribution Solutions
- Storage Solutions
- Special Purpose Appliance Building

Security & Compliance

- Server & Network Security Setups
- Security Testing, Audit and Compliance
- Incident Response
- Managed Security Service

Atola Insight

That's all you need for data recovery.

Atola Technology offers *Atola Insight* – the only data recovery device that covers the entire data recovery process: *in-depth* **HDD diagnostics**, **firmware recovery**, **HDD duplication**, and **file recovery**. It is like a whole data recovery Lab in one Tool.

This product is the best choice for seasoned professionals as well as start-up data recovery companies.

Emphasized features at a glance:

- Automatic in-depth diagnostic of all hard drive components
- Automatic firmware recovery and ATA password removal
- Very fast imaging of damaged drives
- Imaging by heads
- Case management
- Real time current monitor
- Firmware area backup system
- Serial port and power control
- Write protection switch

Atola TECHNOLOGY

Visit atola.com for details

How to use

Socat and Wireshark

for Practical SSL Protocol Reverse Engineering?

Secure Socket Layer (SSL) Man-In-the-Middle (MITM) proxies have two very specific purposes. The first is to allow a client with one set of keys to communicate with a service that has a different set of keys without either side knowing about it. This is typically seen as a MITM attack but can be used for productive ends as well. The second is to view the unencrypted data for security, educational, an reverse engineering purposes.

For instance, a system administrator could set up a proxy to allow SSL clients that don't support more modern SSL methods or even SSL at all to get access to services securely. Typically, this involves having the proxy set up behind your firewall so that unencrypted content stays within the confines of your local area.

Being able to analyze the unencrypted data is very important to security auditors as well. A very large percentage of developers feel their services are adequately protected since SSL is being used between the client and the server. This includes the idea that if the SSL client is custom closed source software that the protocol will be unbreakable and therefore immune to tampering. If you're investing your companies funds using a service that could easily be subject to tampering then you may end up with a nasty surprise. Lost funds perhaps or possibly having your account information publicly available. This article focuses on using an SSL MITM proxy to reverse engineer a simple web service. The purpose of doing so will be to create your own client that can interact with a database behind an unpublished API. The software used will be based on the popular open source software Socat as well as the widely recognized Wireshark. Both are available on most operating systems.

Lets get started!

We will be reverse engineering a LiveJournal client called LogJam which supports SSL connections to the LiveJournal API servers. Since this article is purely educational we don't mind getting some experience using the LiveJournal API which already public and LogJam which is a free and open source project.

Prerequisites

- Install Socat – Multipurpose relay for bidirectional data transfer: *http://www.dest-unreach.org/socat/*
- Install Wireshark – Network traffic analyzer: *http://www.wireshark.org/*
- Install OpenSSL – Secure Socket Layer (SSL) binary and related cryptographic tools: *http://www.openssl.org/*
- Install TinyCA – Simple graphical program for certification authority management: *http://tinyca.sm-zone.net/*
- Install LogJam – Client for LiveJournal-based sites: *http://andy-shev.github.com/LogJam/*

Generating a false SSL certificate authority (CA) and server certificate

The API domain name for LiveJournal is simply www.livejournal.com and any SSL compliant client software will require the server certificate to match the domain when it initially connects to the SSL port of the server.

An SSL CA signs SSL certificates and is nothing more than a set of certificates files that can be used by tools like OpenSSL to sign newly gener-

ated certificates via a *certificate signature request* (CSR) key that is generated while creating new server certificates. The client simply needs to trust the certificate authority public key and subsequently the client will trust all server certificates signed by the certificate authority private key.

Generating a certificate authority

Run `tinyca2` for the first time and a certificate authority generation screen will appear to get you started (Figure 1).

It doesn't matter what you put here if you don't plan on keeping this certificate authority information for very long. The target server at LiveJournal.com will never see the keys you are generating and they will stay completely isolated to your testing environment. Be sure to remember the password since it will be required for signing keys later on.

Select *Export CA* from the *CA* tab and save a *PEM* version of the public CA certificate to a new file of your choosing.

Generating a server certificate

Click on the *Requests* tab in TinyCA and then the *New* button that will help us create a new certificate signing request and private server key (Figure 2).

The common name must be *www.livejournal.com*. The password can be anything and we will be removing it when we export the key for use.

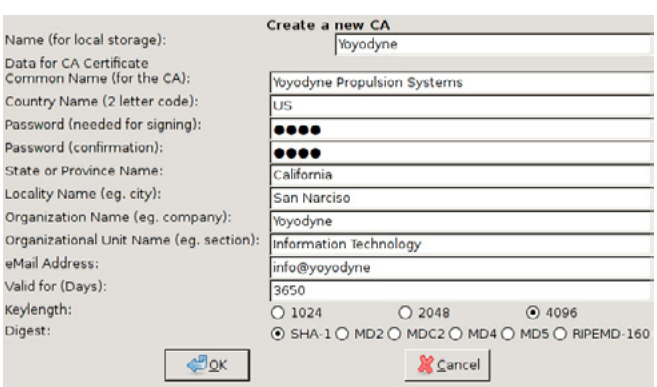


Figure 1. *TinyCA new certificate authority window*

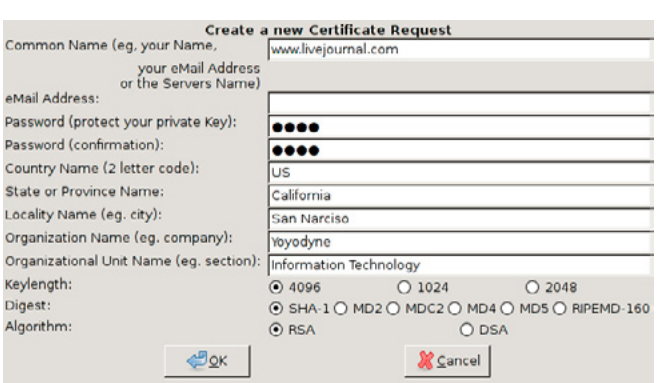


Figure 2. *TinyCA new certificate request window*

Under the *Requests* tab there is now a certificate named *www.livejournal.com* that needs to be signed. Right click and select *Sign Request* and then *Sign Request Server*. Use the default values to sign the request.

Now there will be a new key under the *Key* tab now. Right click on it and select *Export Key* and you'll be presented a new dialog (Figure 3).

As seen in the figure you want to select *PEM (Key)* as well as Without *Passphase (PEM/PKCS#12)* and *Include Certificate (PEM)*. Doing so will export a PEM certificate file that contains a section for the certificate key as well as the certificate itself. The PEM stanard allows us to store multiple keys in a single file.

Congratulations, you now have a perfectly valid key for *https://www.livejournal.com* as long as the web server running the site is under your own control and uses the server key you've generated. Trusting the key is the tricky part.

Allow logjam to trust the certificate authority

So we have to dig in a bit to understand what SSL Certificate trust database LogJam will be using. Most Linux based GTK and console programs rely on OpenSSL which has it's own certificate authority database that is very easy to add a new certificate to.

In Debian/GNU Linux the following will install your new Yoyodyne CA certificate system wide: Listing 1.

Now LogJam as well as programs such as wget, w3m, and most scripting languages will trust all keys signed by your new CA.

Using Socat to proxy the stream and hijacking your own DNS

Socat is basically a swiss army knife for communication streams. With it you can proxy between protocols. This includes becoming an SSL aware server and proxying streams as an SSL aware client to another SSL aware server

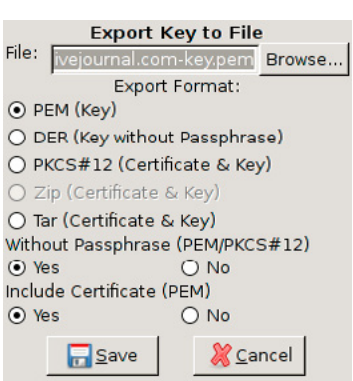


Figure 3. *TinyCA private key export window*

Set up your system and start up socat

Since we should aim for transparency we will need to intercept DNS requests for *www.livejournal.com* as well so that our locally operated proxy running on port `443` on `IP 127.0.2.1` is in the loop.

First, we will need to know the original IP of *www.livejournal.com*:

```
spencersr@bigboote:~$ nslookup www.livejournal.com
                8.8.8.8
Server:     8.8.8.8
Address: 8.8.8.8#53
Non-authoritative answer:
Name: www.livejournal.com
Address: 208.93.0.128
```

Bingo! Now add the following line to `/etc/hosts` near the other IPv4 records:

```
127.0.2.1 www.livejournal.com
```

Now lets do a test run by listening on port 443 (HTTPS) and forwarding to port 443 (HTTPS) of the real *www.livejournal.com*:

```
spencersr@bigboote:~$ sudo socat -vvv \ OPENSSL-
LISTEN:443,verify=0,fork,key=www.livejournal.com-
keyem,certificate=www.livejournal.com-key.pem,
cafile=Yoyodyne-cacert.pem \
OPENSSL:208.93.0.128:443,verify=0,fork
```

Simple enough. Browsing to *https://www.livejournal.com* with w3m and wget should work sucessfully now and a stream of random encrypted information will be printed by socat.

Listing 1. *Install Yoyodyne CA certificate*

```
spencersr@bigboote:~$ sudo mkdir /usr/share/ca-certificates/custom
spencersr@bigboote:~$ sudo cp Yoyodyne-cacert.pem \ /usr/share/ca-certificates/custom/Yoyodyne-
                cacert.crt
spencersr@bigboote:~$ sudo chmod a+rw \
/usr/share/ca-certificates/custom/Yoyodyne-cacert.crt
spencersr@bigboote:~$ sudo dpkg-reconfigure -plow ca-certificates -f readline \ ca-certificates
                configuration
------------------------------
  ...
Trust new certificates from certificate authorities? 1
  ...
This package installs common CA (Certificate Authority) certificates in /usr/share/ca-certificates.
Please select the certificate authorities you trust so that their certificates are installed into
/etc/ssl/certs. They will be compiled into a single /etc/ssl/certs/ca-certificates.crt file.
  ...
  1. cacert.org/cacert.org.crt
  2. custom/Yoyodyne-cacert.crt
  3. debconf.org/ca.crt
  ...
  150. mozilla/XRamp_Global_CA_Root.crt
  151. spi-inc.org/spi-ca-2003.crt
  152. spi-inc.org/spi-cacert-2008.crt
  ...
(Enter the items you want to select, separated by spaces.)
  ...
Certificates to activate: 2
  ...
Updating certificates in /etc/ssl/certs... 1 added, 0 removed; done.
Running hooks in /etc/ca-certificates/update.d....
Adding debian:Yoyodyne-cacert.pem
done.
```

Chaining two socat instances together with an unencrypted session in the middle.
So far so good! Now we need to have socat connecting to another socat using standard TCP4 protocol in order to view the unencrypted data. This works by having one socat instance listening on port 443 (HTTPS) and then forwarding to another socat on port 8080 (HTTP) which then forwards on to port 443 (HTTPS) of the real *www.livejournal.com*.

Listing 2. *Socat terminal*

```
> 2012/08/29 00:10:27.527184  length=209
                  from=0 to=208
POST /interface/flat HTTP/1.1\r
Host: www.livejournal.com\r
Content-Type: application/x-www-form-
                  urlencoded\r
User-Agent: http://logjam.danga.com; martine@
                  danga.com\r
Connection: Keep-Alive\r
Content-Length: 23\r
\r
> 2012/08/29 00:10:27.566184  length=23
                  from=209 to=231
ver=1&mode=getchallenge< 2012/08/29
                  00:10:29.551570  length=437
                  from=0 to=436
HTTP/1.1 200 OK\r
Server: GoatProxy 1.0\r
Date: Wed, 29 Aug 2012 08:10:56 GMT\r
Content-Type: text/plain; charset=UTF-8\r
Connection: keep-alive\r
X-AWS-Id: ws25\r
Content-Length: 157\r
Accept-Ranges: bytes\r
X-Varnish: 904353035\r
Age: 0\r
X-VWS-Id: bil1-varn21\r
X-Gateway: bil1-swlb10\r
\r
auth_scheme
c0
challenge
c0:1346227200:656:60:xxxxxx:xxxxxxxxxxxx
expire_time
1346227916
server_time
1346227856
success
OK
```

Socat instance one:

```
spencersr@bigboote:~$ sudo socat -vvv \
OPENSSL-LISTEN:443,verify=0,fork,
key=www.livejournal.com-key.pem,certificate=
www.livejournal.com-key.pem,cafile=Yoyodyne-
                    cacert.pem \
TCP4:10.1.0.1:8080,fork
```

Socat instance two:

```
spencersr@bigboote:~$ sudo socat -vvv \
TCP-LISTEN:8080,fork \
OPENSSL:208.93.0.128:443,verify=0,fork
```

Load up LogJam and the socat instances will start printing out the stream to the terminal (Listing 2).

Hurray! You should be dancing at this point.

But wait, I mentioned using Wireshark before didn't I?

Using Wireshark to capture and view the unencrypted stream.

Now it's time for the easy part. I'm going to assume that you are comfortable capturing packets in Wireshark and focus mainly on the filtering of the capture stream.

Since by default Wireshark captures all traffic we should set up a capture filter that only listens for packets on port 8080 of host 127.0.2.1 (Figure 4).

Once LogJam is run packet will start streaming in while Wireshark is recording (Figure 5).

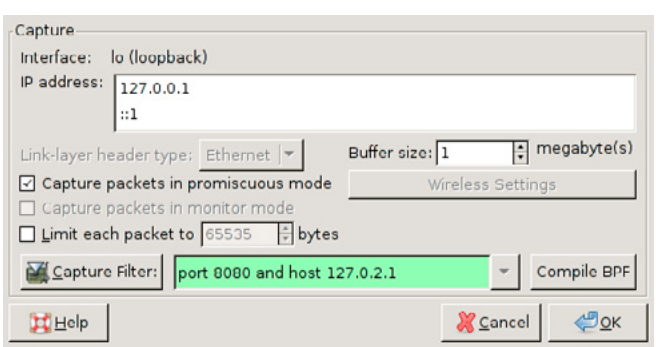


Figure 4. *Wireshark lo (loopback) interface capture window with capture filter*

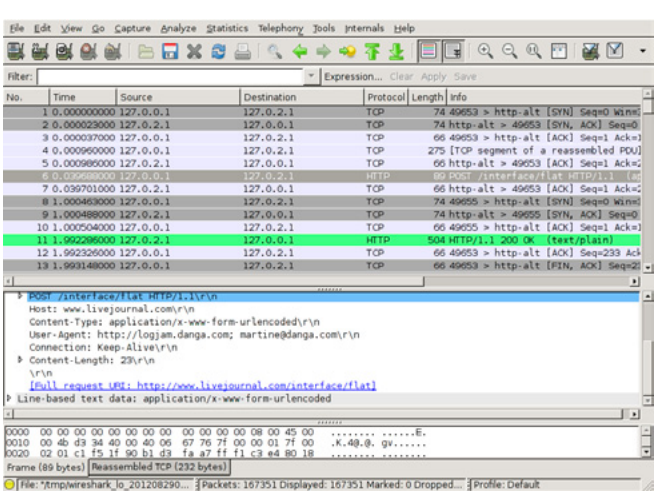


Figure 5. *Wireshark with captured unencrypted packets*

What now?

This articles is about viewing unencrypted data in an SSL session. Whatever your reverse engineering goal is SSL is less of an obstacle now.

How can SSL be secure then if this method is so simple?

SSL and all of the variations of digests and ciphers contained within it are pretty reliably secure. Some of the major areas this article focused on was the ability to fool a client by having the ability to trust a new certificate.

If you are interested in securing your site or client software against this sort of spying I recommend not using an SSL certificate authority keyring or trust database that is easily modified by the user. Including an SSL server certificate in client software ,encrypted and protected by a hard coded key somewhere in the binary, and requiring it for use on SSL connections using a hardened socket library will dramatically cut down on the looky-loo factor.

Conclusion

Thanks to how simple it is to add certificate authorities to most browsers, mobile devices, and custom client software it's a trivial matter to pull back the curtain on SSL encrypted streams with the right tools.

Remember to thank your open source hacker friends.

SHANE R. SPENCER

Shane R. Spencer is based out of Anchorage Alaska and has over 10 years of system administration and programming experience. Many of his projects are Python based and interface with external services that provide no usable API and communicate over HTTPS only.

Infosec Technologies

Reducing risk through technical excellence

Solutions:

- ✓ Two Factor Authentication
- ✓ Email and Web Security
- ✓ Endpoint Security
- ✓ Mobile Device Management
- ✓ Wireless Security
- ✓ Data Governance
- ✓ Secure Remote Access
- ✓ Perimeter Security
- ✓ Intrusion Detection & Prevention
- ✓ Secure Infrastructure

Technology alone cannot solve today's security challenges, but by applying the right mix of technology and services to solve even the most complex security challenges, we are able to reduce both cost and business risk.

Infosec Technologies provides impartial advice and expert technical support that can help you secure your IT infrastructure and achieve your business goals.

About Infosec Technologies:

Infosec Technologies is a UK based, award winning supplier of information security solutions. We have delivered over five hundred projects in the last seven years and have partnerships with both established and new security vendors.

We are dedicated to researching and testing new and innovative technologies to provide our clients with ever stronger, more resilient and agile security products and services.

Our clients span every business sector; from government to pharmaceuticals, financial to ISP, retail and charity. Extensive experience in the design, implementation and support of security and infrastructure solutions allows us to meet specific requirements whilst still maintaining the highest levels of customer service and technical support.

Our technical excellence, focus on customer service and flexible approach ensures we are ready to be your trusted security advisors.

Contact us today for expert advice and support:

Phone: +44 (0)1256 397790
Email: sales@infosectechnologies.com
Website: www.infosectechnologies.com

jscrambler
protect your code

Modern websites, which use Web 2.0 and AJAX, often generate HTML and JavaScript code on the fly. This means that standard static code analyzers cannot fully scan the source code and locate client-side JavaScript issues, since the source code itself does not yet include the entire HTML and JavaScript code.

We used a sample group of 675 websites, including all 500 of the Fortune 500 companies, plus 175 handpicked websites including IT security companies, web application security companies, social networking sites and other popular websites. "*Each application was tested for two main client-side JavaScript issues: DOM-based Cross-site scripting, and open redirects, a vulnerability which allows a malicious attacker to force the victim's browser to automatically redirect to a site he/she owns, and which can be used for Phishing purposes. Our research found that of the 675 websites analyzed, 98 (14.5 percent) were infested with DOM-based Cross site scripting and open redirects* (Figure 1).[1]

1 ftp://public.dhe.ibm.com/common/ssi/ecm/en/raw14252usen/RAW14252USEN.PDF

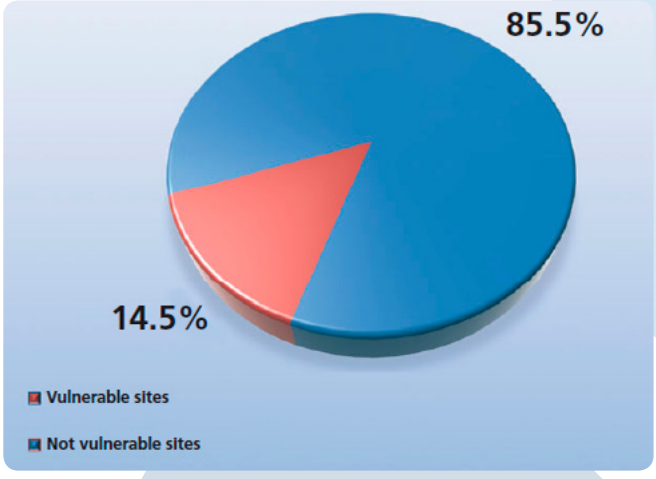


Figure 1. *Percentage of sites vulnerable to client-side JavaScring issues*

Here, the question how I can protect JavaScript code arises. Web Application has to live with JavaScript and it will never be 100% secure. However, there is a known method to protect your JavaScript: source code obfuscation. There are some tools available on market which provide a degree of obfuscation which gives you a bit comfort that your intellectual property (source code) is protected and that it will not be stolen or reused by anyone else in the market.

JScrambler Overview

JScrambler is a JavaScript obfuscator that performs all sorts of complex stuff for your code; it transforms your code into a human-incomprehen-

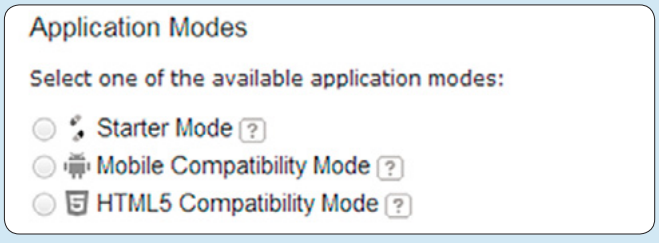


Figure 2. *Shows the application mode of JScrambler*

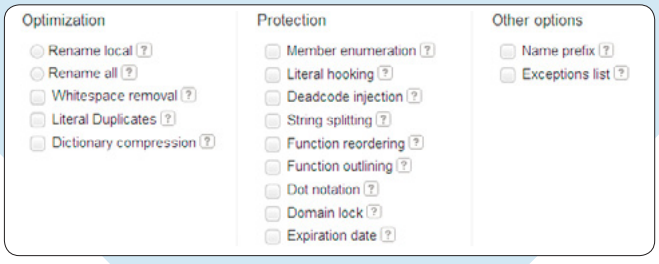


Figure 3. *Shows functionality you can use to achive transformation from protection point of view*

sible form, installs all sorts of protection mechanisms and optimizes the code. **Huh – how about the functionality of your code?** ***Yeah – it transforms and protects while maintaining your code functionality.***

How JScrambler Protects your Code?

I would say if you are looking for a solution to optimize and, at the same time, protect your HTML5, Mobile, Web Game or a standard JavaScript application; then JScrambler is the product you are looking for.

Figure 2 shows the application modes available in JScrambler.

JScrambler is a customizable tool which provides a number of techniques / parameters which you can use in your projects to secure your code. What stands out in JScrambler is its flexibility and its focus on code protection. That being said, it manages also to be one of the best tools for compressing your code. It provides a wide set of customizable options to achieve different degrees of protection, as you can see in **Figure 3**.

With JScrambler's source code obfuscation features you can achieve a certain degree of intellectual property protection by hooking literals, splitting strings into smaller pieces and mixing them throughout the code, reordering function calls, or by injecting dead code to misguide static code reviews. It also provides features to enforce your licence agreement by allowing you to lock the code to a domain list, and/or to make the code expire on certain date after which your customer will not be able to execute it. **Figure 4** – Domain Lock Example.

On top of protection, it has as unique feature a proper validation of the code prior to the application of the source code transformations, by detecting parsing errors just like a normal compiler does. It fully supports the latest JavaScript standard EcmaScript-262 v5.1. **Figure 5** shows an overview of your projects and if parsing errors were detected. This can be helpful to the user as it provides some guarantees that the script is functional before transformation.

Figure 4. *Domain Lock Example*

HTML5 obfuscation – *The only one of its kind*

The HTML5 obfuscation feature of JScrambler is right now the only one available on the market.

You can use JScrambler to hide known calls to the browser DOM objects, or HTML5-specific elements like Canvas. **Figures 6 and 7** show an obfuscated HTML5 Canvas example. You can find the code available at *http://webfensive.com/canvas/*.

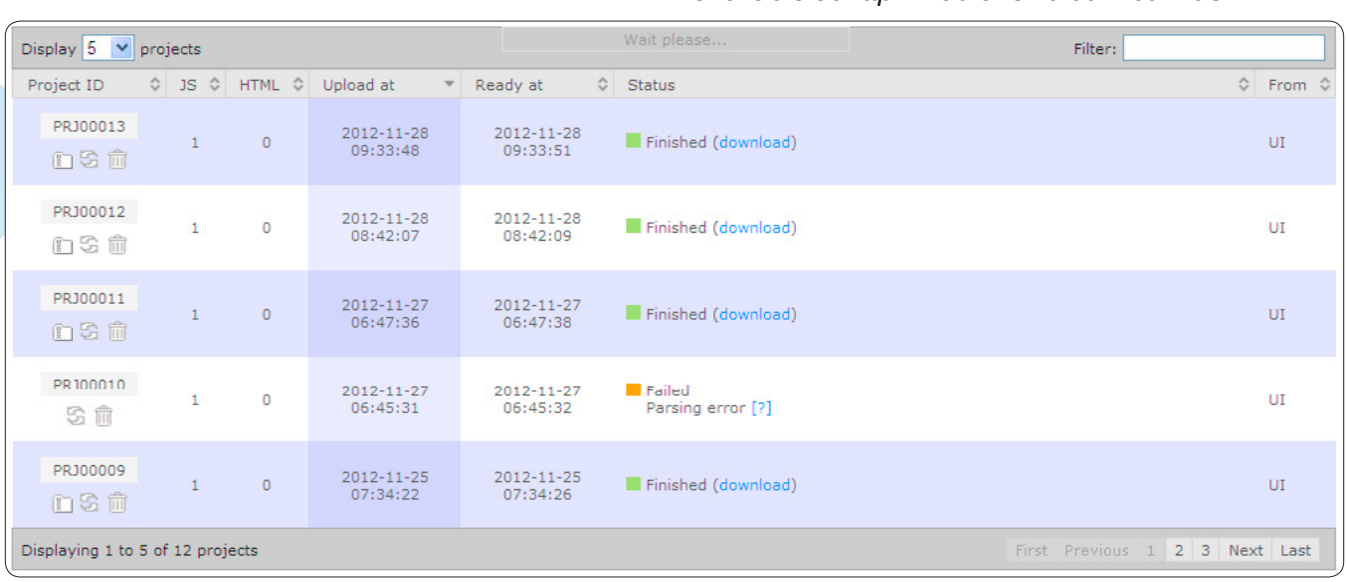


Figure 5. *Shows a quick view of parsing errors*

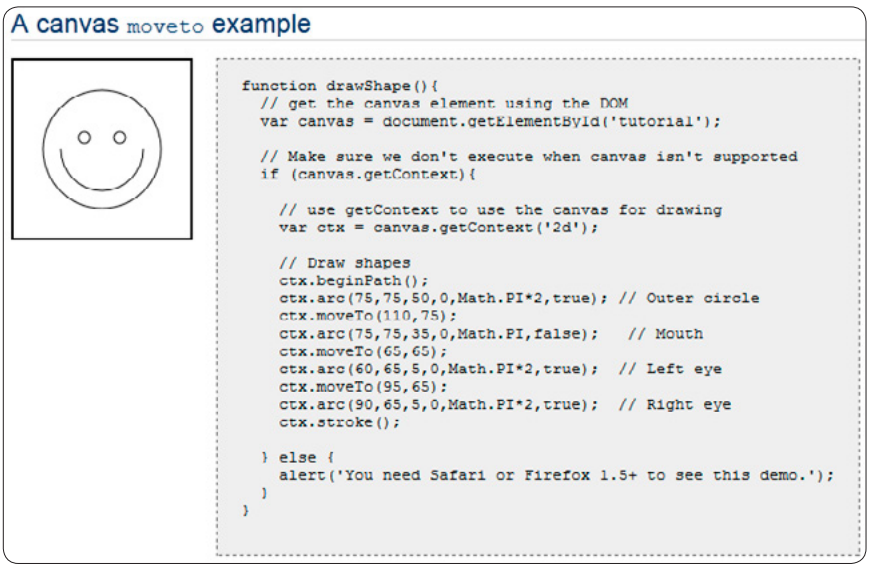


Figure 6. *Before Obfuscation*

Figure 7. *After Obfuscation*

There's also the possibility of adding an exclusion attribute to script tags to make JScrambler ignore code which you don't want it to touch.

Example: <script src="foo.js" jscrambler="ignore"></script>

By applying the aforementioned techniques, you can randomly change the control flow and structure of your JavaScript source code and, at the same time, maintain its functionality.

Conclusion

It is impressively easy and painless to use JScrambler to protect your JavaScript code. JavaScript has been gaining a lot of attention as it is used in different types of applications such as Mobile, HTML5 Canvas and Web Gaming. JScrambler already presents packages tailored to protect those types of applications and it does a good job.

jscrambler
protect your code

RAHEEL AHMAD

Raheel Ahmad, CISSP, is an Information Security Consultant with around 10 years of experience in Information security and forensics.

WEBNETSOFT

Integrated IT Solutions

www.webnetsoft.gr

- ✓ Information Security
- ✓ Network Security
- ✓ Physical Security
- ✓ Software Development
- ✓ IT Services
- ✓ Telecommunications
- ✓ Consulting Services
- ✓ Outsourcing Services

Greece - Attica Glyfada T. +30 213 0024 233 F. +30 211 7807 999
info@webnetsoft.gr

SPTechCon

The SharePoint Technology Conference

March 3-6, 2013 → San Francisco

Get the scoop on SharePoint 2013!

Register Early and SAVE!

The Best SharePoint Training!

Choose from over 90 Classes & Workshops!

Check out these NEW! classes, taught by the industry's best experts!

- How to Install SharePoint 2013 Without Screwing It Up — Todd Klindt and Shane Young
- What IS SharePoint Development? — Mark Rackley
- SharePoint Performance: Best Practices from the Field — Jason Himmelstein
- Creating a Great User Experience in SharePoint — Marc Anderson
- Ten Best SharePoint Features You've Never Used — Christian Buckley
- Understanding and Implementing Governance for SharePoint 2010 — Bill English
- Building Apps for SharePoint 2013 — Andrew Connell
- SharePoint Solutions with SPServices — Marc Anderson
- Lists: Used, Abused and Underappreciated — Wes Preston
- Planning and Configuring Extranets in SharePoint 2010 — Geoff Varosky
- Creating Simple Dashboards Using Out-of-the-Box Web Parts — Jennifer Mason
- Integrating SharePoint 2010 and Visual Studio Lightswitch — Rob Windsor
- Solving Enterprise Search Challenges with SharePoint 2010 — Matthew McDermott
- Getting Stuff Done! Managing Tasks with SharePoint Designer Workflows — Chris Beckett
- SharePoint 2013 Upgrade Planning for the End User: What You Need to Know — Richard Harbridge
- Ten Non-SharePoint Technical Issues That Can Doom Your Implementation — Robert Bogue
- SharePoint MoneyBall: The Art of Winning the SharePoint Metrics Game — Susan Hanley
- Intro to Branding SharePoint 2010 in the Farm and Online — Randy Drisgill and John Ross
- How to Best Develop Requirements for SharePoint Projects — Dux Raymond Sy

Lots more online!

A BZ Media Event

Central Contractor Registration

Follow us: twitter.com/SPTechCon

www.sptechcon.com

LEARN:

- Advancing Computer Science
- Artificial Life Programming
- Digital Media
- Digital Video
- Enterprise Software Development
- Game Art and Animation
- Game Design
- Game Programming
- Human-Computer Interaction
- Network Engineering
- Network Security
- Open Source Technologies
- Robotics and Embedded Systems
- Serious Game and Simulation
- Strategic Technology Development
- Technology Forensics
- Technology Product Design
- Technology Studies
- Virtual Modeling and Design
- Web and Social Media Technologies

www.uat.edu > 877.UAT.GEEK

University of Advancing Technology
UAT
Learn. Experience. Innovate.

You can talk the talk.
Can you walk the walk?

PLEASE SEE **WWW.UAT.EDU/FASTFACTS** FOR THE LATEST INFORMATION ABOUT DEGREE PROGRAM PERFORMANCE, PLACEMENT AND COSTS.